

Cloud based sensor network for environmental monitoring

*Original*

Cloud based sensor network for environmental monitoring / Corbellini, S.; DI FRANCIA, Elisabetta; Grassini, S.; Iannucci, Leonardo; Lombardo, Luca; Parvis, M.. - In: MEASUREMENT. - ISSN 0263-2241. - STAMPA. - 118:(2018), pp. 354-361. [10.1016/j.measurement.2017.09.049]

*Availability:*

This version is available at: 11583/2701371 since: 2021-03-30T11:36:37Z

*Publisher:*

Elsevier B.V.

*Published*

DOI:10.1016/j.measurement.2017.09.049

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

© 2018 ELSEVIER.

*Title: Cloud based sensor network for environmental monitoring*

*Authors:*

*Simone Corbellini*

*Dept. of Electronics and Telecommunications, , Politecnico di Torino, Torino, Italy*

*Elisabetta Di Francia*

*Dipartimento di Scienza Applicata e Tecnologia, Politecnico di Torino, Torino, Italy*

*Sabrina Grassini*

*Dipartimento di Scienza Applicata e Tecnologia, Politecnico di Torino, Torino, Italy*

*Leonardo Iannucci*

*Dept. of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy*

*Luca Lombardo*

*Dept. of Electronics and Telecommunications, , Politecnico di Torino, Torino, Italy*

*Marco Parvis*

*Dept. of Electronics and Telecommunications, , Politecnico di Torino, Torino, Italy*

DOI: 10.1016/j.measurement.2017.09.049

# Cloud based sensor network for environmental monitoring

S. Corbellini<sup>b</sup>, E. Di Francia<sup>a</sup>, S. Grassini<sup>a</sup>, L. Iannucci<sup>b</sup>, L. Lombardo<sup>b</sup>, M. Parvis<sup>b,1,\*</sup>

<sup>a</sup>*Dipartimento di Scienza Applicata e Tecnologia Politecnico di Torino*

<sup>b</sup>*Dipartimento di Elettronica e Telecomunicazioni*

*Politecnico di Torino, Corso Duca degli Abruzzi, 24 - 10129 Torino*

---

## Abstract

This paper describes a complete infrastructure for environmental monitoring which is based on a cloud architecture. The proposed system employs small button-like battery operated sensors which connect through a wi-fi network non è corretto: through wireless link to small receivers capable of routing data to a cloud system. Cloud data can be accessed in real time through smart phones and downloaded for further analyses. Each button-like sensor is capable of measuring temperature and humidity for more than one year without attendance. All data are sent to the receiver and stored inside the button memory so that the system can work also in the absence of power and Internet connection. The architecture takes advantage of the  $\mu$ Panel environment which is conceived to work also with unreliable and slow Internet connections and permits to implement a cloud based infrastructures with the capability to control the sensors even locally and in absence of Internet connectivity. The proposed architecture is currently used in different museums to monitor the local environment inside the display rooms, but the system can be easily extended to open environments without additional costs.

*Keywords:* Environmental monitoring, Cloud based measuring systems.

---

## 1. Introduction

Environment monitoring is an important issue both for guarantying people health and for preserving materials. Ranges of the different quantities as well as the most important quantities of course are related to the specific target, however the infrastructure required for the monitoring is similar in all cases requiring to measure for extended time intervals the quantities of interest in several places at the same time, to concentrate the values in an easy to access

---

\*Corresponding author: marco.parvis@polito.it

repository and to process the acquired data both in real time, for immediate intervention when required, and off-line for quality control and back-trace of any abnormal condition. These general constraints can be easily satisfied by using sensing nodes that connects to a concentrator place by using a wireless transmission to arrange what is usually referred to as a Wireless Sensor Network (WSN).

The idea of using networks of sensors which rely on wireless transmission to arrange distributed measurement system is widely discussed in the measurement and computer science community since the development and diffusion of wireless components [1].

Unfortunately the WSN study is mainly developed in the field of wireless communications and computer networks so that most of the published papers concentrate on the design of wireless sensor networks to optimize data transmission instead of concentrating on accuracy and reliability of the measured quantities.

This work instead tries to discuss the problems related to the measured quantities and to the possibility of being able to reconstruct the history of the environmental conditions the different points were subjected to, regardless of the possible networks issues.

Several papers have been published on this subject that describe different kinds of architectures tailored to different specific environments [2, 3, 4, 5, 6, 7, 8, 9], however even when the application is related to the environment [10, 11] the discussion often does not concentrate on the specific points connected to the main issues and requirements connected with the environment monitoring both on the measurement side and on the fruition side:

- On the measurement side, the system must be able to cope with bad Internet connectivity and/or poor bandwidth: often the monitored environment is not equipped with a stable Internet connection. As an example, in recessed sites the connectivity is often obtained with dedicated GPRS modems that might have to operate with untrustable signal levels (e.g. in the case of basements with very poor signal coverage). In many other cases, the connectivity and the power supply may be intermittent or completely unavailable for long periods.
- The monitored points need to be capillary distributed and the measurements have to be taken for long periods of time without intervention of operators to avoid intolerably increasing the cost of monitoring itself. In addition, the way the measuring points are realized is subjected to several constraints which limit the alternatives in arranging the points themselves. In some cases the monitoring systems have to be capable of working in the absence of power supply, either due to an absence of mains, like in recessed sites, or due to the necessity of avoiding cabling to the sensors like as an example inside museum environments. The requirement of avoiding cabling also impacts on the way the measurements are delivered preventing the use of common analog or digital solutions. In several cases the points

need to be noninvasive to avoid impacting also visually on the monitored environment and this often require using sensors with small dimension [12]. Since this type of WSN is often required to work also in the absence of power supply, a way to store the measurements in a safe place and to deliver them when possible has to be implemented.

- The monitored data may have to be processed requiring remarkable computing power either to combine the measurements coming from several points in order to model complex realities, or to forecast dangerous or abnormal conditions. The monitored data can be either used to trigger alarms or for long term quality assurance and event back-trace. In the first scenario a solution is required which delivers the measurements in either in real-time or in quasi real-time. In the second scenario a reliable memory system is required that can be accessed to perform analyses and a-posteriori studies is required.
- A flexible solution is required to access the monitored data by the different actors, not only when they are in front of a computer at their office, but also when they are around so that a rapid intervention can be performed without the necessity of arranging a continuous and costly surveillance team. In particular, a way to monitor and control the sensors locally through smart-phones has to be implemented in case of Internet unavailability.

All these requirements lead to an infrastructure which basically can be easily arranged in the form of a conventional WSN [8, 15, 16, 17], but which also requires specific design of all elements, in particular: the cloud architecture, the sensing nodes and the devices which collects the node data. Recently, different solutions have been developed to easily implement cloud-based architectures suitable to manage measurement systems with different complexity, as an example the solutions based on Cloudino [13] or Azure Cloud [14]. However, they do not address the issues connected with the need of providing the local control of the measurement sensors through smart-phones without Internet, or with very limited connectivity and available power.

## 2. The Proposed System Architecture

An architecture which is capable of satisfying all the requirements described in previous section can be obtained implementing a three-layer structure which makes use of an intermediate cloud-based storage and processing layer. The proposed architecture is depicted in fig. 1.

- The back-end sensor layer composed of small, low-power wireless-connected, battery-operated sensors (LPW-Sensors) which can be distributed where the measurements have to be acquired. The idea behind this approach is to arrange small devices which are powered by batteries and which can

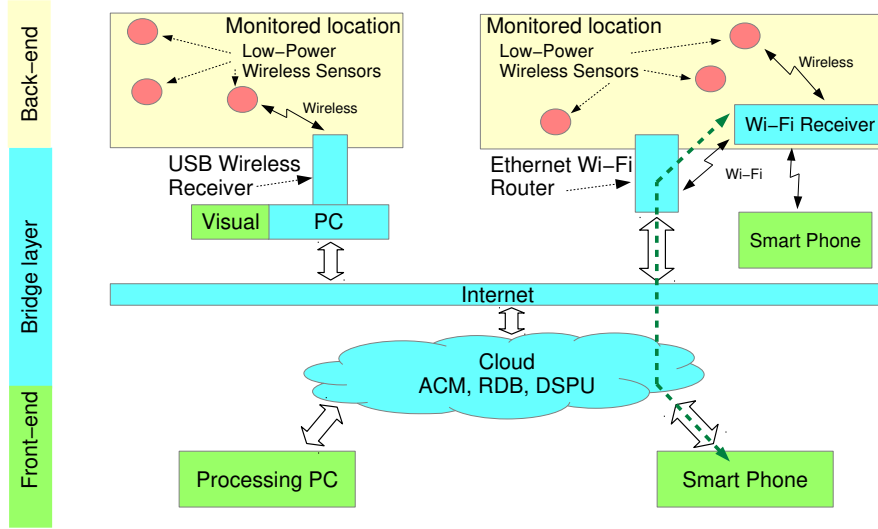


Figure 1: The three-level architecture of the proposed system. The architecture is centered on a cloud with three main elements: Access Control Manager (ACM), Real-time Data Bridge (RDB) and Data Storage and Processing Unit (DSPU).

be deployed without the necessity of specific cabling. Of course having measuring devices which have to be able to work for an extended period of time, usually one year or more and without cabling, implies the necessity of an extremely low energy requirement and this prevents the use of conventional wi-fi connection systems like the ones, as an example, commercially used for smart-phones. In addition, the processing power inside these nodes needs to be low and this requires a solution to move most of the processing capabilities to somewhere else. According to the possibility the other layers stop working, as an example due to an interruption of the power supply, the back-end nodes have to be equipped with enough memory to be able to store all measurements until they can be safely delivered to the subsequent layers.

- The bridge layer, composed of the devices which receive the measurements from the back-end sensors and route them to a cloud infrastructure and of the cloud infrastructure. This bridge layer is designed to collect the measurements generated within the back-end sensor layer and route them to the Internet. The devices belonging to this layer requires a not negligible energy and some computing power to deal with all the requirements connected to the Internet access and therefore have to be connected to a power supply at least for part of their life. Splitting this layer from the acquisition one, let users to arrange only few receiver points and greatly relax the constraints connected to the presence of power supply and ca-

bling. In addition, if some form of local storage is added to the back-end connected part of this layer, the system can be made robust both with respect to the Internet continuity and to the power supply continuity. The cloud infrastructure let putting in this layer also most of the processing power required to manage also large amounts of data lessening the requirements of the front-end and therefore enabling in that layer also the use of low computation power devices such as many low-cost smart-phones. In addition, the cloud system can take care of the data security, enabling only specific user to access selected data according the an access control list, and employing secure encrypting transmission protocols.

- The front-end layer where data are eventually delivered to the final users either to their smart-phones or to their computers. This layer interfaces directly to the users and can be composed of Internet connected PCs and of smart phones that can be used for ubiquitous monitoring of remote sites as well as for timely delivering of alarms and other messages which require immediate attention. This way, if the cloud layer is properly designed, it is possible to perform several post processing activities, but also to be capable of dealing with abnormal conditions regardless of time a location of the responsible agents.

By using this architecture it is possible to relax the computing power requirement of both the back-end and the front-end, concentrating the burden of pre-processing all the measurements in the intermediate bridge layer. In addition, the data transfer between the back-end, where the measurements are taken, and the bridge can be obtained in different ways, that can be tailored according to the specific conditions.

### 3. Back-end sensor design

The back-end design involves arranging a measuring system which is capable of working for an extended period of time without the necessity of being attended and/or connected to a power source and capable of providing a local data storage to overcome any transmission failure. Fig. 2-A shows a generic block diagram of an architecture suitable for the described situation. The sensor is connected to a microcontroller that gets the measurements at specific intervals and delivers them to an ad-hoc wireless transmitter designed to maintain the energy requirements low. A battery powers the entire system and is designed to provide the energy for the whole sensor life. The figure also shows the memory where the measurements can be stored. As told, this memory plays an important role in the sensor since it can both ensure that the measurements can be accessed also at a later time for quality control, but also warrants the measurements are not lost even in the case the wireless connection cannot be activated for any reason to deliver the measurements.

The main issue in this structure is the required sensor life without replacing the battery and its size, which often is required to be small. These constraints

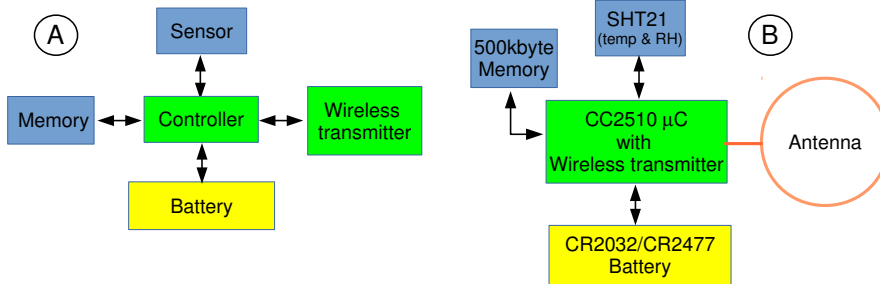


Figure 2: The sensor architecture (A) and a possible implementation (B) based on a CC2510  $\mu C$ .

are of course in contrast and in any case require an optimized low-power design. In addition, depending on the specific kind of sensor to be employed, the node life and or minimum dimension can greatly change.

As an example, fig. 2-B shows a sensor implementation which allows for an extremely long node life and is capable of monitoring temperature and relative humidity. In this implementation the sensor core is a CC2510, a low-power low-cost System on Chip (SoC) made by Texas Instrument, which provides both the wireless transmitter section and computing capabilities, although limited to the power of an 8051 micro-controller. The CC2510 has a cost of less than 3\$ for 1k pieces and contains flash and EEPROM where the sensor firmware can be stored. In addition, the CC2510 contains all the blocks required to drive digital sensors and external memories as well as the wireless radio which is used with a proprietary protocol which minimizes the energy requirement. The radio is capable of a transmission power of 1 dBm; the radio range depends on the used antenna and on the environment, however distances in excess of 10 m can be easily achieved also by employing very small antennas

The sensing element is a Sensirion SHT21 device, a sensor capable of measuring temperature with a typical uncertainty of  $\pm 0.3$  °C and the relative humidity with a typical uncertainty of  $\pm 2\%$ . The sensor delivers data in 16 bit format, i.e. it requires  $B_s = 4$  bytes per measurement. Considering that a measurement interval of 15 minutes is usually satisfactory in environmental monitoring, the memory required in one day is therefore:

$$M = B_s \frac{86400}{S} = 4 \cdot \frac{86400}{900} = 384 \text{ bytes/day} \quad (1)$$

Since a sensor operative life of 3 years is satisfactory in most of the cases, the memory 25Q40 has been installed. This memory provides a storage capability of 500 kBytes which corresponds to about 1300 days, always when measuring every 15 min, i.e. more than 3.5 years.

The back-end node power consumption is related to measurement interval and to the capability of the different components to be put in sleep mode between the measurements



Device	Standby current	Standby Charge in 15min	On current	Charge per transmission	Charge per day
CC2510	0.3 $\mu$ A	0.27 mQ	20 mA	0.4 mQ	64 mQ
SHT21	0.8 $\mu$ A	0.72 mQ	300 $\mu$ A	30 $\mu$ Q	70 mQ
25Q40	$\sim 0$	$\sim 0$	20 mA	0.4 mQ	38 mQ
Total		$\sim 1$ mQ	40.3 mA	0.8 mQ	170 mQ

Table 1: Expected energy consumption of the described back-end node when it is programmed to measure every 15 min

The CC2510 has the capability of being put in sleep mode, still maintaining active the timer used to pace the measurements, with an energy requirement of about 0.3  $\mu$ A. The Sensirion SHT21 can be put in sleep mode as well with an energy requirement of the order of 0.5  $\mu$ A. The memory, if put into sleep mode, would require about 1  $\mu$ A, but the circuit has been designed to let the CC2510 to turn it completely off. This way the total current required during the sleep phases is of only about 0.8  $\mu$ A. The current requirement when the SoC is transmitting data can increase up to more than 25 mA, however, the typical transmission time required to deliver one set of measurements is of the order of 20 ms and this greatly limit the total consumption.

Despite the power consumption of each element can be derived from their datasheets, the estimation of the total power consumption during real operations can result quite difficult due to the existence of many operative states. Also the direct measurement of the current consumption is not straightforward: the CC2510 system on chip quickly switches from long low-power modes to short periods characterized by high current peaks due to CPU activity and to radio data transmission and reception. For this reason a measuring set-up has been arranged indirectly to measure the energy required in each operative state: a large capacitor (i.e. 1000  $\mu$ F) is connected in parallel to an external battery that powers the sensor. To measure the current consumption of a specific state, the battery is disconnected and the voltage across the capacitor is monitored. From the voltage drop and the time interval the mean current can be easily obtained: table 3 reports the charge and corresponding mean current of the main operative states.

With this level of consumption, by using a simple lithium battery like the CR2032, which has a capacity of 220 mAh corresponding to about 790 C a theoretical sensor life in excess of 12 years could be obtained. Of course such a value does not take the both the battery self-discharge and the energy reduction effect due to a use at low temperatures, so the actual back-end node life is lower. For this reason, the CC2510 measures the actual voltage value provided by the battery so that a planned node replacement can be decided when the voltage

drops below 2.7 V. Should a longer life can be required, a battery with a higher capacity like the CR2477 can be used.

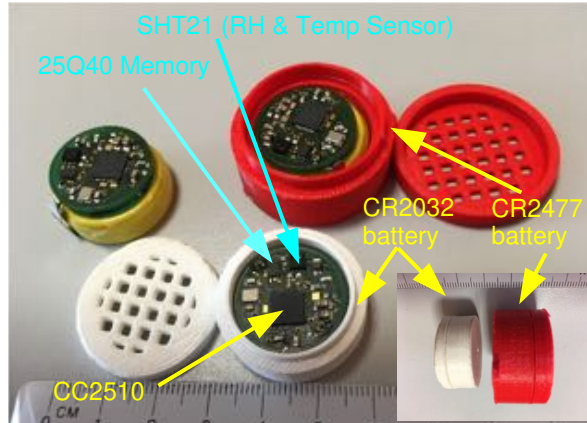


Figure 3: Picture of two prototypes of nodes for measuring temperature and relative humidity. Depending on the battery size the nodes have different size and expected life.

Fig. 3 shows an image of some prototypes of the described temperature and relative humidity sensors. The picture shows two different prototypes. The smaller ones employ the CR2032 lithium battery and have a diameter of about 20 mm, a thickness of only about 15 mm. The larger nodes employ the CR2477 lithium battery and have a diameter of about 26 mm, a thickness of 15 mm. In both cases a 3D printer has been used to realize the case containing the sensors. This way it is possible to easily change the case color according to the specific requirements. A protecting cap lets the air to reach the sensor while providing a safe enclosure. The antenna is obtained with a thin circular path around the sensor printed circuit and provides a transmission capability of more than 10 m in normal environments.

Despite several miniaturized and cheap sensors are available on the market none of them exhibits all the required features. In particular the capability to send data wirelessly and simultaneously record a large amount of data. Even looking for miniaturized data-logger, the memory capability is usually limited to a few thousands of measurements. As an example the *Micro-T* [18] Temperature logger by *Phase IV Engineering Inc*, which is declared as the world's smallest sealed data logger, is just half the size of the designed sensor but can store less than 4000 measures. In addition the device does not embed an active radio and the battery can provide less measurements with respect to the designed sensor. Another example is the miniaturized commercial data logger *DS1921 Thermochron iButton* [19] by *Maxim Integrated*, which is quite popular but can store only 2048 measurements and does not embed the radio as well.

## 4. Bridge layer design

The bridge layer is the key part of the architecture as most of the data processing and storage is performed within this layer. The bridge between data acquisition and users is composed of two separate operations: data receive from the back-end and cloud processing and storage.

### 4.1. Wireless data receivers

The environmental monitoring can be required in different places, however basically two situations can be conceived. If the power supply is available within about 10 m with respect to the back-end sensors and some form of Internet connection is present, the data receiver can implement a continuous data bridge to the cloud, enabling a real-time “*on-line*” monitoring and therefore a timely delivering of the measurements and or of their real-time processed data to the user. In some cases, to save power data can be also collected in the receiver memory and sent in quasi “real-time”, taking into account that delay of several minutes or even hours can be easily acceptable in environment monitoring. On the contrary, if either power supply or Internet connections are not present, such real time connection cannot be obtained and the bridging to the cloud is obtained only “*off-line*” at sporadic intervals, upon user intervention. In both cases the presence of data memory within the back-end nodes ensures that the measurements can be taken and preserved so that it is always possible to trace the environment conditions.

#### 4.1.1. Off-line scenario

When the power supply is not available, i.e. in the “*off-line*” scenario, it is possible to employ a two-step procedure to get the measurements to the cloud: in a first phase a wireless receiver equipped with a large memory and a large battery can be positioned within 10 m from the nodes and let in place to download the measurements. In a second phase the receiver is taken to a location where Internet is available and used to push all the acquired measurements to the cloud. Since all measurement data are stored inside the back-end nodes, a single data receiver can be used to collect the measurements of several nodes also located at large distance one from the other, simply by moving the receiver in different locations.

A convenient way to arrange the “*off-line*” kind of receiver in the described scenario is to use a device capable of using and decoding the proprietary CC2510 protocol and interfacing it to a PC via USB. This way it is not required to provide the receiver with either memory or battery by taking advantage of the PC resources. Fig. 4 shows an example of receiver arranged by using a simple CC2511EMK dongle available off the shelf from Texas Instruments.

This component can be easily programmed to route the wireless packets exchanged with the back-end to the USB so that the back-end can be programmed on the fly and their data received and stored on the PC. When the PC is moved to a location where the Internet is available, all data are pushed to the cloud and



Figure 4: Example of a small USB receiver suitable for operations in the absence of power supply. The dimensions are  $3\text{ cm} \times 4.5\text{ cm}$

made available to the other users. In addition, stored data can be analyzed locally, if required, to observe the environment evolution between two subsequent downloads.

#### 4.1.2. *On-line scenario*

When the power supply is available, i.e. in the “*on-line*” scenario, it is possible to arrange a receiver which allows real-time operations to be obtained. A CC2511 SoC is again the easiest solution to deal with the data transmitted by the back-end but another micro-controller has to be used to deal with the Internet access. Several solutions can be arranged for this operation, however the use of a Wi-Fi link seems the most suitable and flexible one. Making reference to the already described implementation, two different receivers have been arranged according to this choose: the first one is based on a tiny Arduino-like board equipped with the ESP8266 Wi-Fi chip (fig 5, left); the second one is based on the compact Edison Intel module (fig. 5, right). The Arduino based solution has the advantage to be very cheap and low power enough to work for a few days with a lithium battery still keeping active the back-end wireless link, thus continuing receiving the measurements also in case of temporary mains supply interruptions. The Edison based solution, instead, can provide more computational power running the Linux operating system on an Atom processor, which may be useful when complex pre-processing is required to be performed locally or when the employed sensors generate a large volume of data to be managed (e.g. for data compression).

In both cases the back-end data are pushed to the cloud in real-time, or quasi real-time, so that it is possible to perform a continuous monitoring of the environment. In addition, thanks to the Wi-Fi interface, the receiver can also be configured to let authorized users to access the back-end data on the fly

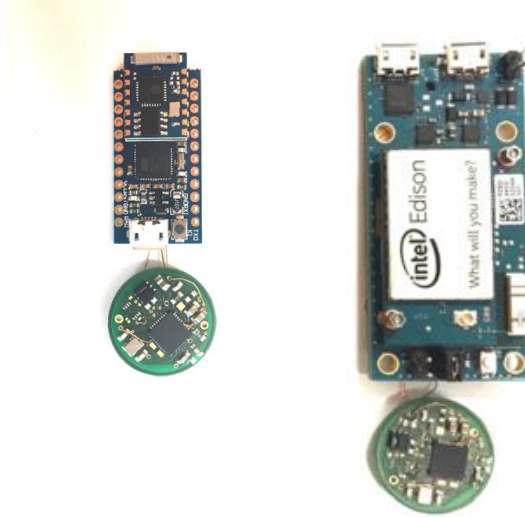


Figure 5: Example of smalls Wi-Fi Internet-Enabled receivers suitable for operating also with sporadic absence of power supply (left) and for intensive pre-processing elaboration (right).

and reconfigure its functioning (e.g. for changing the measuring interval of the wireless sensors).

The access can be easily obtained by means of simple smart-phones if a solution like  $\mu Panel$  [20] is used. This solution, which is based on a small firmware to be installed on the receivers and on a free *App* running on the smart-phone, is conceived to implement control systems but can be used to quickly deploy measurement systems as well.

The  $\mu Panel$  approach relies on the transmission of events and graphics through small data packets based on a specific compressed HTML language (HCTML), and therefore is suitable also for slow and/or not reliable connections, and let network installers to quickly configure simple panels with buttons, graphs and alarms which get displayed on the smart-phones. The  $\mu Panel$  environment seamless integrates in the cloud environment, in particular, the smart-phones can either connect directly to the receiver, in case of Internet unavailability and if the user is near to them, or to the regular Internet over the cloud connection in all the other places.

#### 4.2. Cloud storage and processing

The cloud is used to provide a world-wide connection to smart-phones, in order to let users to monitor in real time the environment when a connection is available also in the monitored site, and to manage data off-line in all cases. The cloud consists of three conceptually separated components: the Access Control Manager (ACM), the Real-Time Data Bridge (RDB) and the Data Storage and Processing Unit (DSPU).

- The ACM manages the access of both wireless receivers and mobile devices to the cloud, granting the access only to those elements which are known and allowed; when a wireless receiver is powered up, it connects to the Internet and tries to join the cloud through the ACM. Each receiver identifies itself by transmitting over a secure link its unique ID number stored in its ROM. During this phase, the ACM can receive from the receiver also the passwords for the pairing with mobile devices.

The ACM also exports some application program interfaces (APIs) that permit the receivers to generate push notifications to be sent to the registered smart phones. This permits to deliver real time alarms to all the administrators whose mobile device has successfully registered at least once to the ACM and their passwords are still valid. The ACM function is independent of the nature of the measurements to be processed by the system, in fact, all measurement data are redirected by the ACM to a specific Data Storage and Processing Cloud (DSPU), associated with the receiver, and which knows how to deal with the measurements collected by the wireless receiver.

- The RDB works as data bridge allowing receivers and connected mobile devices to exchange data like during a local direct Wi-Fi connection. This function allows administrator and curators to display and interact with the receiver's graphical panel from every location with Internet connectivity. Also the RDB functions are independent of the nature of the measurements.
- The DSPU, which is the only element in the cloud that has to know the nature of the measurements collected by the sensors, receives the data from the wireless receivers and stores them into a suitable database.

The DSPU is able to process the measurements collected into the database and to export them through the front-end to both curators or public. In addition, since the DSPU can access all the generated measurements, it can perform a grid-like data processing in order to detect more complex critical situations, e.g. alarms arising from measurements collected in adjacent rooms by different receivers. In case of critical situation the DSPU can notify the alarms by emails or by asking the ACM to send instant push notifications to the involved curators and administrators. Eventually, the DSPU provide some API for exporting also the raw data stored into the DB in order to let the users perform additional data processing and presentation.

The current cloud implementation takes advantage of the  $\mu Panel$  environment as well, which already provides both the ACM and the RDB elements. Only the DSPU has to be implemented according to the kind of measurements collected by the system.

## 5. Front end

In the realized implementation the front-end has the aim of providing the graphical interface for both administrating the system and analyzing the collected measurements. According to the block diagram of fig. 1 three different kinds of user interface can be used:

- A local visualization provided directly on the computer employed for receiving the measurements in the “off-line” scenario. In the described implementation this interface is created in Visual Basic and allows the operator to control the correctness of the received data and also reconfigure the sensors (e.g. the measurement interval). A graphical representation allows the user also to immediately detect critical situations that may require an immediate local intervention.
- A more useful interface consists in the visualization of the receiver’s graphical control panel directly on the administrators’ smart-phones. In this case the user can connect from everywhere to a specific wireless receiver and display its graphical interface. In the current implementation according to the employed  $\mu Panel$  architecture, the graphical interface is described in HCTML language: fig. 6 shows an example of receiver’s control panel where the users can check the status (left) and last measurements of each sensors (right). Through the receiver’s panels it is also possible to re-configure the sensors parameters, such as the measurement and transmission intervals.
- The highest level of user interface is provided by the cloud DSPU unit, which is able to process all the gathered measurements. In this case the data are exported by means of common web-pages in which authorized users can query the DSPU data-base and display the acquired measurements for a given location and period of time and request the cloud computers to perform specific processing without the necessity to install and develop software on the curator computers.

## 6. Results

A complete implementation of the proposed system has been installed inside a museum where several showcases are present. Five sensors have been installed inside five different showcases, and an additional sensor has been installed in the room containing the showcases. A low-Power wireless receiver like the one shown in fig. 5-left has been used to permit a continuous monitoring thanks to

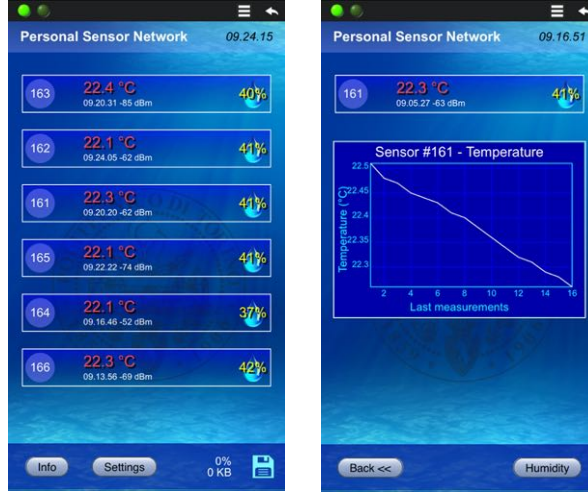


Figure 6: Screen shot of a wireless receiver’s graphical panel displayed on a smart-phone screen showing the sensors’ status (left) and the plot of last temperature measurements generated by one sensor (right). In the right bottom corner of the status panel it is shown the amount of data currently stored into the receiver’s memory and still waiting to be uploaded to the cloud.

a WiFi network obtained by means of a GPRS modem. The system has been started in July 2016 and is regularly working and collecting data over nearly one year. About 10 Mbytes have been generated in this period. Thanks to the modem, the collected data are available for instantaneous monitoring on the conservator smart-phone as shown in fig. 6. The collected data are also stored inside the cloud DSPU and can be retrieved, processed and plotted by means of the Processing clients: as an example, fig. 7 shows some different ways the data recorded by the six sensors over the last 9 months can be obtained by the cloud.

Fig. 7-A shows the average values of temperature and relative humidity regarding all the sensors. In this case the cloud applies a filter to remove the circadian variations in order to highlight the temperature decreases with the winter season down to average values of 15 ° while the humidity increases to more than 50%. Fig. 7-B instead shows the raw results of over the same period under-sampled to keep the size to the data transferred to client low. In this case it is easy to observe the large humidity changes measured by the sensor which is outside the show-cases, which has serial number 164. These traces highlight an abnormal behavior, which is highlighted by the red circle, showing that something strange happened in December 2016 when the humidity recorded by in the showcases monitored by some of the sensors suddenly increase. In this case it is possible to ask the cloud more detailed data as shown in Fig. 7-C where an expansion of the abnormal interval is shown. From theses data



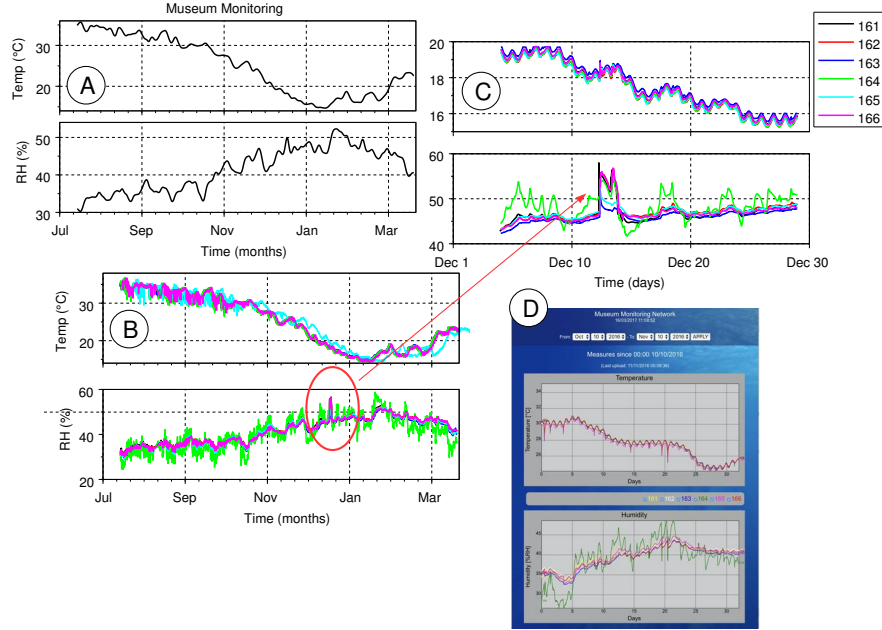


Figure 7: Different kinds of data that can be obtained by accessing the cloud DSPU. A: the average recorded values filtered by the cloud to remove the circadian cycle; B: the raw values of the different sensors; C: an expansion of the time interval when one of the showcases has been opened; D: the screen-shot one can obtain directly by using the web page.

appears that the showcases monitored by sensors 161 and 166 were opened and their content subjected to the environment humidity which reached almost 60%. After two days they were sealed again fortunately during a time interval where the humidity was well below 50% thus re-creating a safe environment. Eventually, Fig. 7-D shows an example of the simplest screen shot readily available simply by using a browser and regarding one month of data: the sensor installed in the room, and identified with the serial 164 clearly shows the humidity changes of about 10% during the day.

These results demonstrate the capability of the proposed architecture to be easily deployed and used for museum monitoring. Other solutions based on more power-computing elements, such as raspberry boards and other embedded computers, could be employed as well, however the small size and compactness of the proposed architecture's elements proved to be ideal to be employed in exposed areas.

## 7. Conclusions

The problem of environmental monitoring inside museums as well as inside many other types of buildings where a controlled environment is required is

becoming more and more requested.

In most cases the constraints related to the invasiveness of the available systems prevent the use of off-the-shelf solutions, and specific solutions have to be developed that have to be tailored for each specific application. This paper describes a flexible architecture which employs a cloud processing and a possible implementation based on the *μPanel* environment that permits an extremely easy and fast deployment of complex monitoring systems.

The proposed solution permits to have real time clients based on personal smart-phones for unconstrained and ubiquitous monitoring, but also processing clients to show the monitoring history and to perform complex analyses. The real time system to be used with the smart-phones employs an extremely compact transmission scheme, is designed to work also in case of networks with limited throughput and support the push notifications to timely deliver alarm to the users. Despite the described implementation focuses on temperature and humidity monitoring, the possibility to add low-power GAS sensors [21, 22], such as those based on optical fibers [23], is being investigated for future implementations.

## References

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “Wireless sensor networks: a survey”, *Computer Networks*, vol. 38, pp. 393–422, 2002
- [2] Fu-To Lin et al - A Self-Powering Wireless Environment Monitoring System Using Soil Energy, *IEEE Sensors Journal*, vol 15, no. 7, 2015 pp. 3751-3758
- [3] A. Kumar, G. P. Hancke - Energy Efficient Environment Monitoring System Based on the IEEE 802.15.4 Standard for Low Cost Requirements, *IEEE Sensors Journal* vol 14, no. 9, 2014, pp. 2557-2566
- [4] C. Micheloni, P. Remagnino, H. Eng, J. Geng - Intelligent Monitoring of Complex Environments, *IEEE Intelligent Systems*, vol 25, no. 3, 2010, pp. 12-14
- [5] J. Albusac et al - Monitoring Complex Environments Using a Knowledge-Driven Approach Based on Intelligent Agents, *IEEE Intelligent Systems*, Vol. 25, no. 3, 2010, pp. 24-31
- [6] G. Zhongwen, et al, “IMA: An Integrated Monitoring Architecture With Sensor Networks”, *IEEE Tr. on Instrumentation and Measurement*, vol. 61, no.5 pp. 1287-1295, 2012
- [7] J. Schmalzel; F. Figueroa; J. Morris; S. Mandayam; R. Polikar, “An architecture for intelligent systems based on smart sensors”, *IEEE Tr. on Instrumentation and Measurement*, vol. 54, no. 4, pp. 1612-1616, 2005
- [8] A. Carullo; S. Corbellini; F. Ferraris; M. Parvis; A. Vallan, “Integration of low computational power nodes in a GRID measurement architecture”,

IEEE Instrumentation and Measurement Technology Conference IMTC 2007, pp. 1-5, 2007

- [9] B. Gedik; L. Liu, “A scalable peer-to-peer architecture for distributed information monitoring applications”, IEEE Tr. on Computers, vol. 54, no. 6, pp. 767-782, 2005
- [10] M. T. Lazarescu “Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications”, IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 3, no. 1, pp. 45-54, 2013
- [11] A. Boubrima; W. Bechkit; H. Rivano, “Optimal WSN Deployment Models for Air Pollution Monitoring”, IEEE Transactions on Wireless Communications, vol. 16, no. 5, pp. 2723-2735, 2017
- [12] National Park Service: “Comparing Temperature and Relative Humidity Dataloggers for Museum Monitoring” Conserve O Gram Series, N.3/3 Sept. 2011, available at <http://www.nps.gov/museum/publications/conservoogram/03-03.pdf>, last checked 2015, Sept. 20
- [13] Cloudino. Available at <http://cloudino.io/>, last checked 2017 August 9.
- [14] Azure Cloud Services. Available at <https://azure.microsoft.com/en-us/services/cloud-services/>, last checked 2017 August 9.
- [15] Wen-Tsai Sung, Chia-Cheng Hsu - Intelligent environment monitoring system based on innovative integration technology via programmable system on chip platform and ZigBee network IET Communications, vol 7, no. 16, 2013 pp. 1789-1801
- [16] E. M. Petriu; T. E. Whalen; R. Abielmona; A. Stewart - Robotic sensor agents: a new generation of intelligent agents for complex environment monitoring IEEE Instrumentation & Measurement Magazine, vol 7, no. 3, 2004, pp. 46-51
- [17] D. J. Kim et al - A Multisensor Mobile Interface for Industrial Environment and Healthcare Monitoring, IEEE Transactions on Industrial Electronics, vol. 64, no. 3, 2017 pp. 2344-2352
- [18] The Micro-T data logger. Available at <http://www.phaseivengr.com/wireless-technologies/about-rfid-wireless-data-loggers/about-hf-nfc-rfid-wireless-data-loggers/> , last checked 2017, Aug, 10.
- [19] The iButton. Available at <https://www.maximintegrated.com/en/products/ibutton/ibuttons/thermochron.cfm> , last checked 2017, Aug. 10.
- [20] The  $\mu$ Panel system. Available at <http://www.miupanel.com>, last checked 2017, Aug. 10.

- [21] C. M. Grzywacz, “Monitoring for Gaseous Pollutants in Museum Environments”, The Getty Conservation Institute, 2006, available at [http://www.getty.edu/conservation/publications\\_resources/pdf\\_publications/pdf/monitoring.pdf](http://www.getty.edu/conservation/publications_resources/pdf_publications/pdf/monitoring.pdf), last checked 2015, Sept. 20.
- [22] D. Camuffo et. al, “Environmental monitoring in four European museums”, *Atmospheric Environment*, Vol.35, supp.1, 2001, pp. S127-S140
- [23] S. Corbellini; M. Parvis; S. Grassini; L. Benussi; S. Bianco; S. Colafranceschi; D. Piccolo; “Modified POF sensor for gaseous hydrogen fluoride monitoring in the presence of ionizing radiations”, *IEEE Transactions on Instrumentation and Measurement*, Vol. 61, Issue 5, 2012, pp. 1201-1208.